

# JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Készítette: **Nagy Gábor**

Neptunkód: **H194QS**

## A feladat leírása:

A feladatban témája, hogy egy céget hogyan tudunk adatbázisban eltárolni és annak alkalmazottjait, részlegeit, illetve projektjeit fogom számontartani.

Egy nagy multinacionális cég nyilvántartása nagyon nehéz, mondhatni igazából lehetetlen hiszen nem lehet számontartani hol mennyien dolgoznak, mi az elérhetőségük stb.

A céghez közvetlenül három adattal fog rendelkezni, cégid, cégnév, illetve a cégcím tulajdonsággal, amely egy összetett tulajdonság, így az irányítószámból, városból, utcából és a házszámból épül fel.

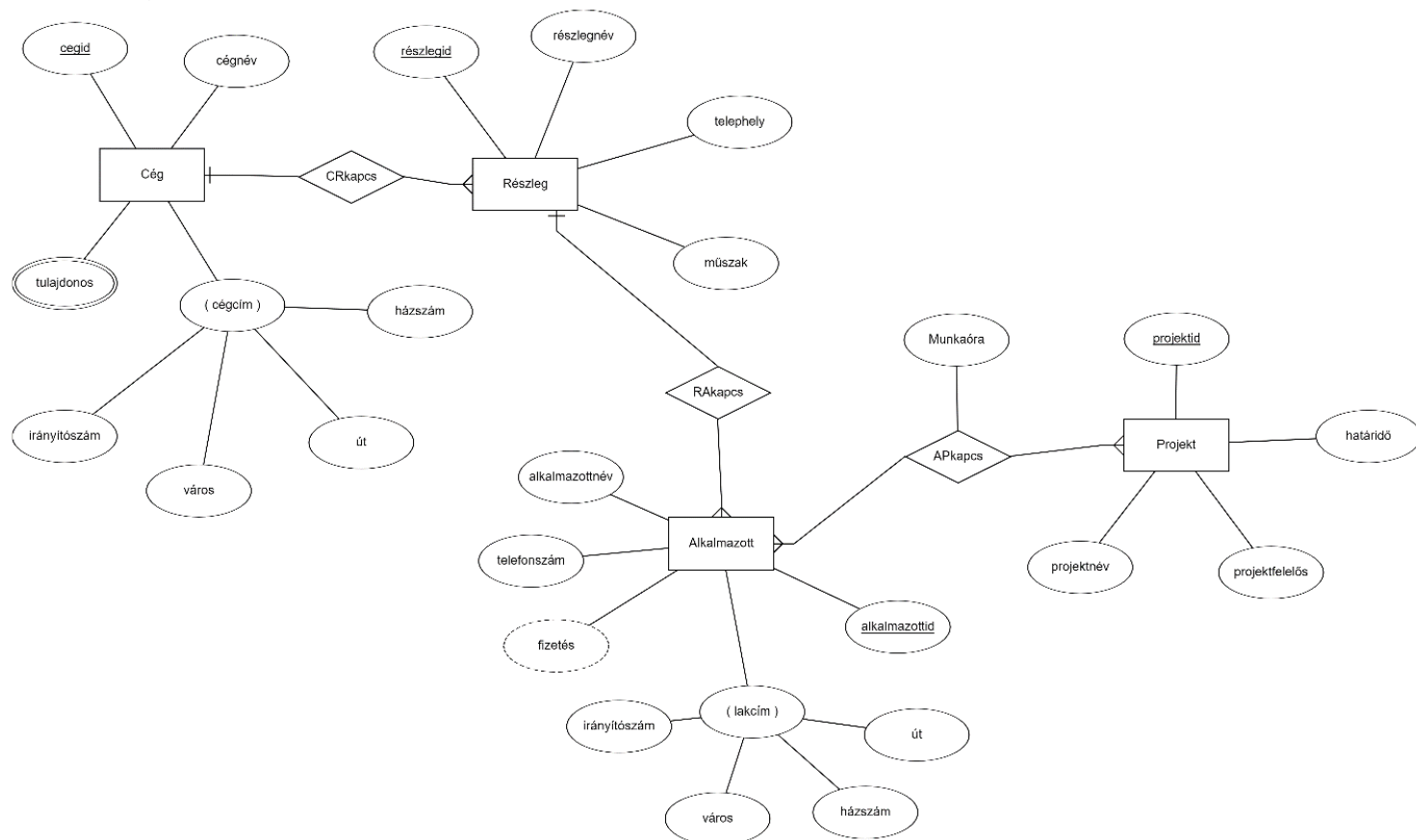
Fontos meghatároznunk a céghez tartozó részlegeket, amelynél öt tulajdonságot fogunk tárolni, a részlegid-t, a cégid-t részlegnevet, telephelyét és hogy mennyi műszakba dolgoznak ott az alkalmazottak.

A részleghez tartozik minimum egy alkalmazott, de akár több alkalmazott is ezért öt adatot fogunk tárolni az alkalmazottakról. Ezek az adatok az alkalmazottid, részlegid, alkalmazottnév, telefonszám, és egy lakcím.

Különböző cégeknél különböző alkalmazottakhoz rendelünk projekteket ezért a projektnek is fontos eltárolnunk négy különböző adatot. Ahhoz, hogy könnyen tudjuk azonosítani a projektet ezért fontos eltárolni a projektid-t, projektnevet, projektfelelőst és a határidőt.

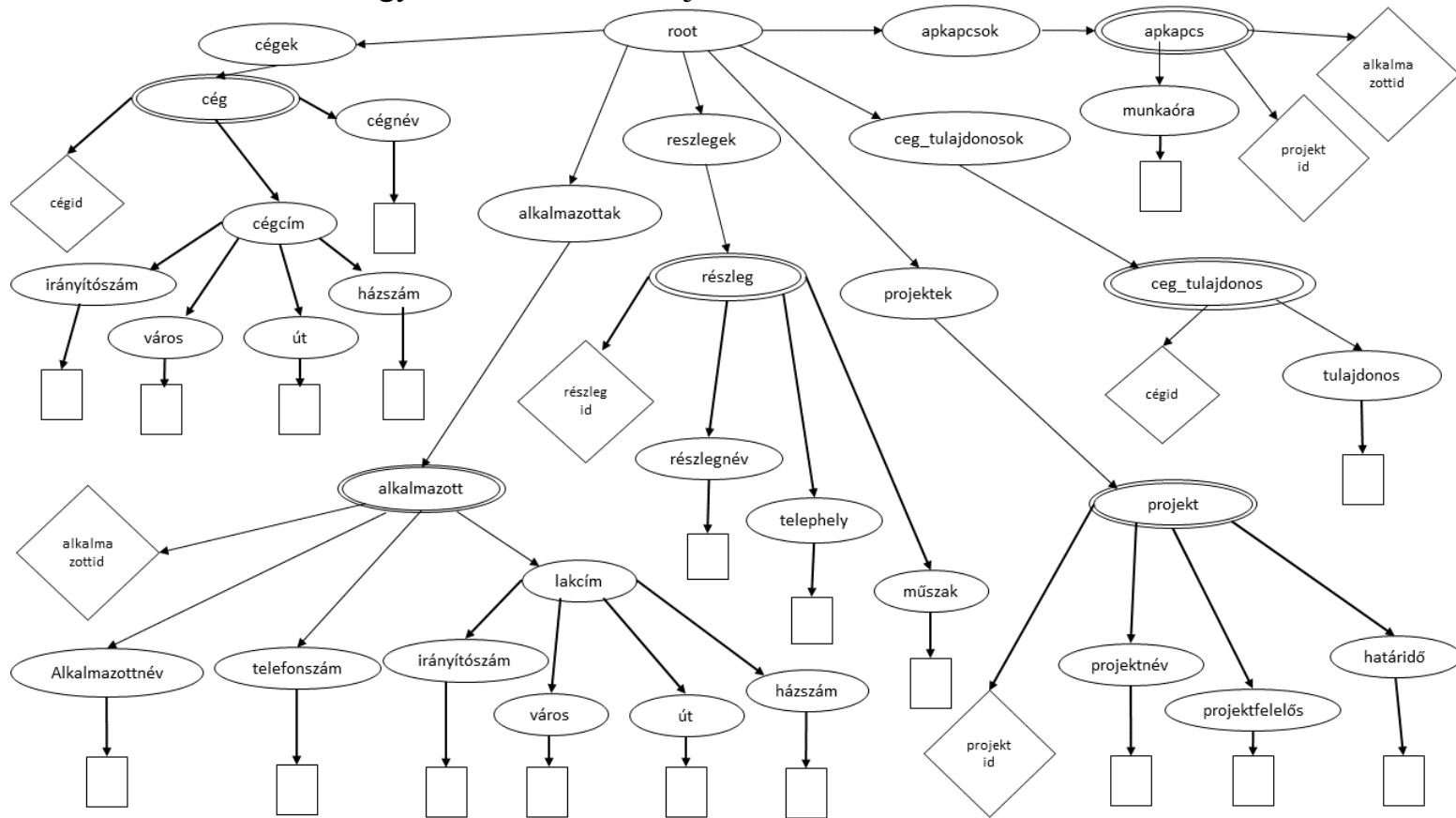
### 1. feladat

#### 1a) Az adatbázis ER modell:



## 1b) Az adatbázis konvertálása XDM modellre:

E-R modellt XDM modellre úgy konvertálhatunk, hogy három féle jelölést használunk. Van az elem, amely az XDM modellben ellipszissel van rajzolva. Van az attribútum, amelyet a rombusz jelöl, illetve van a szöveg, amelyet a téglalap jelöl a képen. Ezeken az építőköveken kívül megtalálható az XDM modellben egy dupla ellipszis, amely az elem többszörös előfordulási lehetőségét határozza meg. Ezenkívül minden egyedhez és N:M kapcsolathoz felvesszük az egyed többszám alakját is.



## 1c) Az XDM modell alapján XML dokumentum készítése:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <alkalmazottak>
    <alkalmazott alkalmazottid="a1" részlegid="r1">
      <alkalmazottnev>Koszor Károly</alkalmazottnev>
      <telefonszam>06302658471</telefonszam>
      <lakcim>
        <iranyitoszam>4515</iranyitoszam>
        <varos>Nincslaki</varos>
        <ut>Kimondhatatlan</ut>
        <hazszam>123</hazszam>
      </lakcim>
    </alkalmazott>
    <alkalmazott alkalmazottid="a2" részlegid="r1">
      <alkalmazottnev>Sáros Sándor</alkalmazottnev>
      <telefonszam>+36202658471</telefonszam>
      <lakcim>
        <iranyitoszam>4500</iranyitoszam>
        <varos>Vanlaki</varos>
        <ut>Kimondható</ut>
        <hazszam>321</hazszam>
      </lakcim>
    </alkalmazott>
    <alkalmazott alkalmazottid="a3" részlegid="r2">
      <alkalmazottnev>Sima Sándor</alkalmazottnev>
      <telefonszam>06302648957</telefonszam>
      <lakcim>
        <iranyitoszam>4517</iranyitoszam>
        <varos>Lehetlaki</varos>
        <ut>Nemlaki</ut>
        <hazszam>258</hazszam>
      </lakcim>
    </alkalmazott>
    <alkalmazott alkalmazottid="a4" részlegid="r2">
      <alkalmazottnev>Kormos Károly</alkalmazottnev>
      <telefonszam>06702658957</telefonszam>
      <lakcim>
        <iranyitoszam>2875</iranyitoszam>
        <varos>Bűnös</varos>
        <ut>Börtön</ut>
        <hazszam>15</hazszam>
      </lakcim>
    </alkalmazott>
  </alkalmazottak>
  <apkapcsok>
    <apkapcs alkalmazottid="a1" projektid="p1">
      <Munkaora>15</Munkaora>
    </apkapcs>
    <apkapcs alkalmazottid="a2" projektid="p1">
      <Munkaora>50</Munkaora>
    </apkapcs>
    <apkapcs alkalmazottid="a2" projektid="p2">
      <Munkaora>50</Munkaora>
    </apkapcs>
    <apkapcs alkalmazottid="a3" projektid="p1">
      <Munkaora>157</Munkaora>
    </apkapcs>
    <apkapcs alkalmazottid="a4" projektid="p2">
      <Munkaora>168</Munkaora>
    </apkapcs>
  </apkapcsok>
  <cégek>
    <ceg cegid="c1">
      <cegnev>Alma Kft.</cegnev>
      <cegcim>
```

```

        <iranyitoszam>8888</iranyitoszam>
        <varos>Miskolc</varos>
        <ut>Nemtudom</ut>
        <hazszam>154</hazszam>
    </cegcim>
</ceg>
<ceg cegid="c2">
    <cegnev>Körte Kft.</cegnev>
    <cegcim>
        <iranyitoszam>1248</iranyitoszam>
        <varos>Budapest</varos>
        <ut>Kínos</ut>
        <hazszam>258</hazszam>
    </cegcim>
</ceg>
</cégek>
<ceg_tulajdonosok>
    <ceg_tulajdonos cegid="c1">
        <tulajdonos>Piros Alma</tulajdonos>
    </ceg_tulajdonos>
    <ceg_tulajdonos cegid="c1">
        <tulajdonos>Zöld Alma</tulajdonos>
    </ceg_tulajdonos>
    <ceg_tulajdonos cegid="c2">
        <tulajdonos>Zöld Körte</tulajdonos>
    </ceg_tulajdonos>
</ceg_tulajdonosok>
<projektek>
    <projekt projektid="p1">
        <projektnev>Kettő forint</projektnev>
        <projektfelelős>Koszor Károly</projektfelelős>
        <hataridő>2020.10.15</hataridő>
    </projekt>
    <projekt projektid="p2">
        <projektnev>Három forint</projektnev>
        <projektfelelős>Sáros Sándor</projektfelelős>
        <hataridő>2025.01.24</hataridő>
    </projekt>
</projektek>
<reszlegek>
    <reszleg reszlegid="r1" cegid="c1">
        <reszlegnev>Préselő</reszlegnev>
        <telephely>Miskolc</telephely>
        <műszak>3</műszak>
    </reszleg>
    <reszleg reszlegid="r2" cegid="c2">
        <reszlegnev>Préselő</reszlegnev>
        <telephely>Budapest</telephely>
        <műszak>1</műszak>
    </reszleg>
    <reszleg reszlegid="r3" cegid="c2">
        <reszlegnev>Préselő</reszlegnev>
        <telephely>Miskolc</telephely>
        <műszak>2</műszak>
    </reszleg>
</reszlegek>
</root>

```

## 1d) Az XML dokumentum alapján XMLSchema készítése:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="root">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="alkalmazottak">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="alkalmazott" minOccurs="1"
                maxOccurs="1000">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="alkalmazottnev" type="xs:string" />
                    <xs:element name="telefonszam" type="telefonTipus" />
                    <xs:element name="lakcim">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element name="iranyitoszam"
                            type="iranyitoszamTipus" />
                          <xs:element name="varos" type="xs:string" />
                          <xs:element name="ut" type="xs:string" />
                          <xs:element name="hazszam"
                            type="xs:positiveInteger" />
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                  <xs:attribute name="alkalmazottid"
                    type="xs:string" />
                  <xs:attribute name="reszlegid" type="xs:string" />
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="apkapcsok">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="apkapcs" minOccurs="1"
                maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="munkaora" type="munkaoraTipus" />
                  </xs:sequence>
                  <xs:attribute name="alkalmazottid"
                    type="xs:string" />
                  <xs:attribute name="projektid" type="xs:string" />
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="cegek">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ceg" minOccurs="1" maxOccurs="100">
                <xs:complexType>
```

```

        <xs:sequence>
            <xs:element name="cegnev" type="xs:string" />
            <xs:element name="cegcim">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="iranyitoszam"
                            type="irányitoszamTipus" />
                        <xs:element name="varos" type="xs:string" />
                        <xs:element name="ut" type="xs:string" />
                        <xs:element name="hazszam"
                            type="xs:positiveInteger" />
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
        <xs:attribute name="cegid" type="xs:string" />
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ceg_tulajdonosok">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="ceg_tulajdonos" minOccurs="1"
                maxOccurs="5">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="tulajdonos" type="xs:string" />
                    </xs:sequence>
                    <xs:attribute name="cegid" type="xs:string" />
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="projektek">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="projekt" minOccurs="1"
                maxOccurs="10">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="projektnev" type="xs:string" />
                        <xs:element name="projektfelelos" type="xs:string" />
                        <xs:element name="hatarido" type="datumTipus" />
                    </xs:sequence>
                    <xs:attribute name="projektid" type="xs:string" />
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="reszlegek">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="reszleg" minOccurs="1"
                maxOccurs="25">

```

```

        <xs:complexType>
            <xs:sequence>
                <xs:element name="reszlegnev" type="xs:string" />
                <xs:element name="telephely" type="xs:string" />
                <xs:element name="muszak" type="muszakTipus" />
            </xs:sequence>
            <xs:attribute name="reszlegid" type="xs:string" />
            <xs:attribute name="cegid" type="xs:string" />
        </xs:complexType>
    </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:key name="Key_alkalmazott">
    <xs:selector xpath="alkalmazott"/>
    <xs:field xpath="@alkalmazottid"/>
</xs:key>
<xs:key name="Key_ceg">
    <xs:selector xpath="ceg"/>
    <xs:field xpath="@cegid"/>
</xs:key>
<xs:key name="Key_projekt">
    <xs:selector xpath="projekt"/>
    <xs:field xpath="@projektid"/>
</xs:key>
<xs:key name="Key_reszleg">
    <xs:selector xpath="reszleg"/>
    <xs:field xpath="@reszlegid"/>
</xs:key>
<xs:keyref name="FKey_ceg_tulajdonos_ceg" refer="Key_ceg">
    <xs:selector xpath="ceg_tulajdonos"/>
    <xs:field xpath="@cegid"/>
</xs:keyref>
<xs:keyref name="FKey_reszleg_ceg" refer="Key_ceg">
    <xs:selector xpath="reszleg"/>
    <xs:field xpath="@cegid"/>
</xs:keyref>

<xs:keyref name="FKey_alkalmazott_reszleg" refer="Key_reszleg">
    <xs:selector xpath="alkalmazott"/>
    <xs:field xpath="@reszlegid"/>
</xs:keyref>
<xs:keyref name="FKey_APkapcs_alkalmazott" refer="Key_alkalmazott">
    <xs:selector xpath="apkapcs"/>
    <xs:field xpath="@alkalmazottid"/>
</xs:keyref>
<xs:keyref name="FKey_APkapcs_projekt" refer="Key_projekt">
    <xs:selector xpath="apkapcs"/>
    <xs:field xpath="@projektid"/>
</xs:keyref>
</xs:element>
<xs:simpleType name="iranyitoszamTipus">
    <xs:restriction base="xs:positiveInteger">
        <xs:minInclusive value="1000" />
        <xs:maxInclusive value="9999" />
    </xs:restriction>

```



```

</xs:simpleType>
<xs:simpleType name="muszakTipus">
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="1" />
    <xs:maxInclusive value="3" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="telefonTipus">
  <xs:restriction base="xs:string">
    <xs:pattern value="[0]{1}[0-9]{2}[0]{1}[0-9]{7}" />
    <xs:pattern value="[+]{1}[0-9]{3}[0]{1}[0-9]{7}" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="datumTipus">
  <xs:restriction base="xs:string">

    <!-- 2000.5.1 ,2000.5.01 -->
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[1-9]{1}[.]{1}[1-9]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[1-9]{1}[.]{1}[1-2]{1}[0-9]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[1-9]{1}[.]{1}[0]{1}[1-9]{1}" />

    <!-- 2000.10.2,2000.10.02 -->
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[1]{1}[1]{0-2}{1}[.]{1}[1-9]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[1]{1}[1]{0-2}{1}[.]{1}[1-2]{1}[0-9]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[1]{1}[1]{0-2}{1}[.]{1}[0]{1}[1-9]{1}" />

    <!-- 2000.05.1 ,2000.05.01 -->
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[1-9]{1}[.]{1}[1-9]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[1-9]{1}[.]{1}[1-2]{1}[0-9]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[1-9]{1}[.]{1}[0]{1}[1-9]{1}" />

    <!-- január 31nap -->
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[1]{1}[1]{.}{1}[3]{1}[0-1]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[1]{1}[1]{.}{1}[3]{1}[0-1]{1}" />

    <!-- február 29nap -->

    <!-- március 31nap -->
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[3]{1}[1]{.}{1}[3]{1}[0-1]{1}" />
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[1]{3}{1}[.]{1}[3]{1}[0-1]{1}" />

    <!-- április 30nap -->
    <xs:pattern
      value="[1-9]{1}[0-9]{3}[.]{1}[4]{1}[1]{.}{1}[3]{1}[0]{1}" />
    <xs:pattern

```

```

        value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[4]{1}[.]{1}[3]{1}[0]{1}" />

<!-- május 31nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[5]{1}[.]{1}[3]{1}[0-1]{1}" />
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[5]{1}[.]{1}[3]{1}[0-1]{1}" />

<!-- június 30nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[6]{1}[.]{1}[3]{1}[0]{1}" />
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[6]{1}[.]{1}[3]{1}[0]{1}" />

<!-- július 31nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[7]{1}[.]{1}[3]{1}[0-1]{1}" />
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[7]{1}[.]{1}[3]{1}[0-1]{1}" />

<!-- augusztus 31nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[8]{1}[.]{1}[3]{1}[0-1]{1}" />
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[8]{1}[.]{1}[3]{1}[0-1]{1}" />

<!-- szeptember 30nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[9]{1}[.]{1}[3]{1}[0]{1}" />
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[0]{1}[9]{1}[.]{1}[3]{1}[0]{1}" />

<!-- október 31nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[1]{1}[0]{1}[.]{1}[3]{1}[0-1]{1}" />

<!-- november 30nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[1]{1}[1]{1}[.]{1}[3]{1}[0]{1}" />

<!-- december 31nap -->
<xs:pattern
    value="[1-9]{1}[0-9]{3}[.]{1}[1]{1}[2]{1}[.]{1}[3]{1}[0-1]{1}" />

    </xs:restriction>
</xs:simpleType>
</xs:schema>

```

## 2.feladat

A feladat egy DOM program készítése az XML dokumentum adatainak adminisztrálása alapján:

### 2a) adatolvasás

```
package hu.dompars.h194qs;

import java.io.File;

public class DOMReadH194QS {

    public static void main(String[] args) {
        NodeList list;
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();
            // fájl beolvasása
            Document document = builder.parse(new File("XMLH194QS.xml"));
            document.getDocumentElement().normalize();
            // Gyökér elem megkeresése
            System.out.println("Root element : " + document.getDocumentElement().getNodeName());
            System.out.println("-----");
            // Aktuális elem meghatározása
            list = document.getElementsByTagName("alkalmazott");

            for (int i = 0; i < list.getLength(); i++) {
                Node node = list.item(i);
                System.out.println("\nCurrent element: " + node.getNodeName());
                // Alkalmazott adatainak kiírása
                if (node.getNodeType() == Node.ELEMENT_NODE) {
                    Element element = (Element) node;
                    System.out.println("Alkalmazott id: " + element.getAttribute("alkalmazottid"));
                    System.out.println("Reszleg id: " + element.getAttribute("reszlegid"));

                    System.out.println("Alkalmazottnév: "
                        + element.getElementsByTagName("alkalmazottnév").item(0).getTextContent());
                    System.out.println("Telefonszám: " + element.getElementsByTagName("telefonszam").item(0).getTextContent());

                    // Alkalmazottan belüli lakcím adatok kiírása
                    Node nodelakcim = list.item(i);
                    if (nodelakcim.getNodeType() == Node.ELEMENT_NODE) {
                        Element elementlakcim = (Element) node;
                        System.out.println("Írányítószám: "
                            + elementlakcim.getElementsByTagName("iranyitoszam").item(0).getTextContent());
                        System.out.println("Város: " + elementlakcim.getElementsByTagName("varos").item(0).getTextContent());
                        System.out.println("Út: " + elementlakcim.getElementsByTagName("ut").item(0).getTextContent());
                        System.out.println("Házszám: " + elementlakcim.getElementsByTagName("hazszam").item(0).getTextContent());
                    }
                }
            }
            // Aktuális elem meghatározása
            list = document.getElementsByTagName("apkapcs");

            for (int i = 0; i < list.getLength(); i++) {
                Node node = list.item(i);
                System.out.println("\nCurrent element: " + node.getNodeName());
                // Apkapcs adatainak kiírása
                if (node.getNodeType() == Node.ELEMENT_NODE) {
                    Element element = (Element) node;
                    System.out.println("Alkalmazott id: " + element.getAttribute("alkalmazottid"));
                    System.out.println("Projekt id: " + element.getAttribute("projektid"));
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```

        System.out
            .println("Munkaóra: " + element.getElementsByTagName("munkaora").item(0).getTextContent());
    }
}
// Aktuális elem meghatározása
list = document.getElementsByTagName("ceg");

for (int i = 0; i < list.getLength(); i++) {
    Node node = list.item(i);
    System.out.println("\nCurrent element: " + node.getNodeName());
    // Cég adatainak kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        System.out.println("Cég id: " + element.getAttribute("cegid"));

        System.out.println("Cégnév: " + element.getElementsByTagName("cegnev").item(0).getTextContent());
        // Cégen belüli cégcím adatok kiírása
        Node nodelakcim = list.item(i);
        if (nodelakcim.getNodeType() == Node.ELEMENT_NODE) {
            Element elementlakcim = (Element) node;
            System.out.println("Irányítószám: "
                + elementlakcim.getElementsByTagName("iranyitoszam").item(0).getTextContent());
            System.out.println(
                "Város: " + elementlakcim.getElementsByTagName("varos").item(0).getTextContent());
            System.out.println("Út: " + elementlakcim.getElementsByTagName("ut").item(0).getTextContent());
            System.out.println(
                "Házzszám: " + elementlakcim.getElementsByTagName("hazzsam").item(0).getTextContent());
        }
    }
}
// Aktuális elem meghatározása
list = document.getElementsByTagName("ceg_tulajdonos");

for (int i = 0; i < list.getLength(); i++) {
    Node node = list.item(i);
    System.out.println("\nCurrent element: " + node.getNodeName());
    // Cég tulajdonos adatainak kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        System.out.println("Cég id: " + element.getAttribute("cegid"));

        System.out.println(
            "Cég tulajdonos: " + element.getElementsByTagName("tulajdonos").item(0).getTextContent());
    }
}
// Aktuális elem meghatározása
list = document.getElementsByTagName("projekt");

for (int i = 0; i < list.getLength(); i++) {
    Node node = list.item(i);
    System.out.println("\nCurrent element: " + node.getNodeName());
    // Projekt adatainak kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        System.out.println("Projekt id: " + element.getAttribute("projektid"));

        System.out.println(
            "Projekt név: " + element.getElementsByTagName("projektnev").item(0).getTextContent());
        System.out.println("Projekt felelős: "
            + element.getElementsByTagName("projektfelelos").item(0).getTextContent());
        System.out
            .println("Határidő: " + element.getElementsByTagName("hatarido").item(0).getTextContent());
    }
}
// Aktuális elem meghatározása
list = document.getElementsByTagName("reszleg");

for (int i = 0; i < list.getLength(); i++) {
    Node node = list.item(i);
    System.out.println("\nCurrent element: " + node.getNodeName());
    //Részleg adatainak kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        System.out.println("Részleg id: " + element.getAttribute("reszlegid"));
        System.out.println("Cég id: " + element.getAttribute("cegid"));

        System.out.println(
            "Részlegnév: " + element.getElementsByTagName("reszlegnev").item(0).getTextContent());
        System.out.println("Telephely: "
            + element.getElementsByTagName("telephely").item(0).getTextContent());
        System.out
            .println("Műszak: " + element.getElementsByTagName("muszak").item(0).getTextContent());
    }
}

} catch (ParserConfigurationException e) {
    e.printStackTrace();
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```

## 2b) adatmódosítás

```
public class DOMModifyH194QS {  
  
    public static void main(String[] args) {  
        NodeList list;  
        try {  
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();  
            DocumentBuilder builder = factory.newDocumentBuilder();  
            // fájl beolvasása  
            Document document = builder.parse(new File("XMLH194QS.xml"));  
            document.getDocumentElement().normalize();  
            // Aktuális elem meghatározása  
            list = document.getElementsByTagName("alkalmazott");  
            for (int i = 0; i < list.getLength(); i++) {  
                Node node = list.item(i);  
                System.out.println("\nCurrent element: " + node.getNodeName());  
                // Alkalmazott adatainak kiírása  
                if (node.getNodeType() == Node.ELEMENT_NODE) {  
                    Element element = (Element) node;  
                    // Alkalmazottid eltárolása egy stringbe  
                    String id = element.getAttribute("alkalmazottid");  
  
                    System.out.println("Alkalmazott id: " + element.getAttribute("alkalmazottid"));  
                    System.out.println("Részleg id: " + element.getAttribute("reszlegid"));  
                    System.out.println("Alkalmazottnév: "  
                        + element.getElementsByTagName("alkalmazottnev").item(0).getTextContent());  
                    System.out.println(  
                        "Telefonszám: " + element.getElementsByTagName("telefonszam").item(0).getTextContent());  
                    // Alkalmazottan belüli lakcím adatok kiírása  
                    Node nodeLakcim = list.item(i);  
                    if (nodeLakcim.getNodeType() == Node.ELEMENT_NODE) {  
                        Element elementLakcim = (Element) node;  
                        // id vizsgálata és ha megegyezik a1-el akkor irányítószám módosítás  
                        if (id.equals("a1")) {  
                            elementLakcim.getElementsByTagName("iranyitoszam").item(0).setTextContent("1111");  
                        }  
                        System.out.println("Irányítószám: "  
                            + elementLakcim.getElementsByTagName("iranyitoszam").item(0).getTextContent());  
                        System.out.println(  
                            "Város: " + elementLakcim.getElementsByTagName("varos").item(0).getTextContent());  
                        System.out.println("Út: " + elementLakcim.getElementsByTagName("ut").item(0).getTextContent());  
                        System.out.println(  
                            "Házszám: " + elementLakcim.getElementsByTagName("hazszam").item(0).getTextContent());  
                    }  
                }  
            }  
            // létrehozza az xml fájlt módosítva  
            Transformer transformer = TransformerFactory.newInstance().newTransformer();  
            Source input = new DOMSource(document);  
            Result output = new StreamResult(new File("XMLH194QSModify.xml"));  
            transformer.transform(input, output);  
        } catch (ParserConfigurationException e) {  
            e.printStackTrace();  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
}
```